CICE Consortium

Progress Report

October 2016 - February 2018, updated June 2018

February 24, 2018

Since the CICE Consortium's kick-off workshop in October 2016, staff supported by the seven participating institutions (5 agencies) focused on implementing the Consortium's processes and preparing its first software release in an open development environment. This required particular attention to building testing capabilities, developing online resources including documentation, and turning the column physics modules from the CICE model into an independent software package, Icepack. The Consortium's decisions have drawn on information and experience from the DOE/E3SM, NPS/RASM, and NCAR/CESM models, the Navy's Verification and Validation expertise, NOAA's proprietary (NWS) and open (GFDL) software experiences, and the Canadian team's "shovel-ready" fast-ice parameterization for developing the procedures needed to accomodate community contributions. During this process, we developed management and communication processes, and grew into a cohesive team with strong interactions across the organization. Most of the products of our efforts can be found at https://github.com/CICE-Consortium.

Major accomplishments

- Kick-off workshop and subsequent team building
- Management processes for planning, communicating across the team and with the community, setting priorities, and tracking tasks
- Governance documents addressing intellectual property concerns
- Open development environment at github.com, transitioned from private LANL repository
- Complete scripting system for building and running our models
- Automated testing and test reporting
- Online documentation, including technical information, user guides and general info
- Independent column physics package, Icepack
    - New driver and scripts for developing and testing the column physics
    - Interface redesign for linking the column physics modules with host models
    - Multiple configurations and forcing scenarios for testing
    - Travis CI (continuous integration) for automated testing
    - Documentation extracted from CICE's into Icepack's own collection
    - Icepack v1.0.0 released February, 2018
- CICE model development

    - Quality control testing procedure (developed by NPS)
    - Fast ice parameterization in CICE (developed by Canadians)
    - Updated interface for Icepack's column physics
- Consortium visibility via numerous briefings to various agencies and the public
- Community interactions

These topics are discussed in more detail below, including contributions by the participating agencies and future plans.

Kick-off workshop and team building

The idea of forming a sea ice modeling consortium was first presented in early 2016, at two workshops involving multi-agency discussions of sea ice and climate modeling. Prior to the Consortium's kick-off workshop in October 2016, participants discussed how they might engage in such a consortium. During the workshop, a roadmap for setting up the organization and its key functions was worked out. This included a broad governance model and division of the workload into teams, with representatives from one or two of the participating agencies leading each team.

In addition, several of the agencies planned to hire or contract personnel to work on Consortium projects. In particular, NCAR hired a community liaison (full-time), NRL-Stennis hired a contractor (part-time), and a Software Engineer was also contracted (part-time), initially with support through NCAR and now through NOAA. Although agencies are responsible for ensuring that their teams' work is accomplished, all of the CICE Consortium's personnel collaborate broadly across the teams as needed — their contributions are not restricted to a particular team. This flexibility serves the Consortium well in its ability to deliver quality software products and in its resilience and cohesiveness.

An early disappointment occurred when the UK Met Office pulled out, citing their decision to engage more fully with Europe's NEMO consortium. They had initially volunteered to co-lead the community support team, leaving NCAR with primary responsibility for that team's work. NOAA's contract with the Software Engineer was delayed, but NCAR was able to support him in the interim, a solution that worked well for NCAR, as their newly hired liaison needed time to transition from a previous project. A remaining challenge is identification of an Icepack team co-lead (with GFDL) at LANL. The Consortium's Lead Coordinator has been doing this work with some help from other staff at LANL, but this is necessarily a temporary solution due to other demands on her time (the fundamental reason that the Consortium formed in the first place).

Workshop report	ftp://ftp.cgd.ucar.edu/archive/Model-Data/CICE/documents/workshopreport_final.pdf

Management processes

The Consortium has developed a number of management processes for planning, communicating across the team and with the community, setting priorities, and tracking tasks. Communication is primarily through online tools, including email and github's issue tracking, projects, and pull request tools. Google docs are used for planning and discussions outside of our public web space. The full team meets monthly using webex to share documents (the agendas are on google docs), and subgroups meet as needed. We held two face-to-face meetings in 2017, during conferences that many team members were already attending: CESM in June and AGU in December. In addition, the Lead Coordinator provided briefings to the Executive Oversight Board (March 2017) and to the group of Sponsors (February, June 2017) by teleconference.

The process for making decisions involves writing a planning document, sharing it with the team for comments, reaching consensus (or the Lead Coordinator decides), and finally creating a set of tasks for doing the work (github issues). The Lead Coordinator sets priorities overall for the Consortium, with input from the whole team. She and the Software Engineer are responsible for code reviews. We specify technical leads based on agencies' team responsibilities and solicit help from others as needed. Links to examples:

Sample agenda	https://docs.google.com/document/d/1KRvA5QZgBuVo3h70exBisWKaf6dx9ADCJlkLaYXj5GU/edit#heading=h.xl205zarbjm3
Technical planning document	https://docs.google.com/document/d/1UAY-SxqaeUdB-gGbrbAADfVhE258as-wVD6p34az_dk/edit#heading=h.x6txrleztlcx
Icepack Issues	https://github.com/CICE-Consortium/Icepack/issues
Icepack project boards	https://github.com/CICE-Consortium/Icepack/projects/1 https://github.com/CICE-Consortium/Icepack/projects/2
CICE v6 project board	https://github.com/CICE-Consortium/CICE/projects/2

Governance documents

Governance for the Consortium was broadly outlined before the kick-off workshop and refined thereafter, including formalizing expectations for agency participation and addressing intellectual property concerns. In order to move the CICE code from LANL's subversion repository to an open development environment at github.com, LANL asserted copyright and we instituted a 3-clause BSD license with reference to our distribution policy, which applies to work shared outside the Consortium's repositories. The 4-month copyright-assertion process delayed

technical progress somewhat, time that we used to get governance and other policies in place, plan our github repository structure and documentation, and discuss strategies for testing the codes. The documents linked below were circulated to the Executive Oversight Board (May 2017) and Sponsors (June 2017) before being posted on our github space.

General information	https://github.com/CICE-Consortium/About-Us
Governance	https://github.com/CICE-Consortium/About-Us/blob/master/Governance.pdf
Terms of Reference	https://github.com/CICE-Consortium/About-Us/blob/master/TermsofReference.pdf
Policies and Guidelines	https://github.com/CICE-Consortium/About-Us/blob/master/PoliciesandGuidelines.pdf
3-Clause BSD License	https://github.com/CICE-Consortium/CICE/blob/master/LICENSE.pdf
Code Distribution Policy	https://github.com/CICE-Consortium/About-Us/blob/master/DistributionPolicy.pdf
Software Development Practices	https://github.com/CICE-Consortium/About-Us/blob/master/SoftwareDevelopmentPractices.pdf

Open development environment

We moved the whole CICE trunk, which had formerly been password protected in a LANL subversion repository, to a public repository on github. Each release resides on its own branch so that bug fixes can be incorporated while more substantial development proceeds on the master branch. The entire subversion repository, including all private branches, was archived on the ACME-Climate (DOE E3SM) github space as a private repository. It is no longer in use but code that was not merged into the trunk can be accessed if necessary.

A branch of the original CICE repository containing an intermediate version of CICE, with the column physics separated into a subdirectory, was implemented and thoroughly tested in the Regional Arctic System Model (RASM) and became the founding code for the Consortium. DOE climate modelers adopted the column physics in DOE's new climate model (ACME/E3SM).

The column physics subdirectory was extracted from CICE into its own repository (Icepack), and linked with CICE as a git submodule. A new, simplified driver was written for Icepack, and scripts and tests were developed for both CICE and Icepack. Data sets and large documents are available from a dedicated ftp site provided by NCAR.

The Consortium makes extensive use of github, utilizing its issues tool, Travis CI integration, code reviews, discussion, pull requests and project tracking. There is a clear separation between open development and the master trunk via the pull request process. The Consortium's github space is entirely open for reading. Writing to the master branches is restricted to a few gatekeepers ("owners": Lead Coordinator, Software Engineer, Community Liaison, and others as needed), although anyone can submit modifications to be considered for merging through github's pull request system. Users' forks are also visible to everyone, but writing is limited to the repository owners and other people given specific permission by the users who own the forks. Because an entire repository is cloned to a user's computer, which is not visible to everyone, development can be done privately as long as the code changes are not pushed back to the user's fork on github. However, sharing of private development code is more difficult among people who do not all have access to the local computer; in contrast, it is relatively easy using public forks on github. The user is responsible for keeping his or her own version of the code in sync with the main repository; this is required before any modifications will be merged from the user's fork into the main repository. For these reasons, the Consortium encourages users to merge developments into the main repository in smaller batches that can be evaluated individually, rather than in one large pull request at the end.

Home page	https://github.com/CICE-Consortium
Trunk from subversion repository	https://github.com/CICE-Consortium/CICE-svn-trunk
Archived subversion repository (private, invisible without access permission)	https://github.com/E3SM-Climate/CICE-archive
ftp repository	ftp://ftp.cgd.ucar.edu/archive/Model-Data

Automated build/run/test scripts and test reporting

Scripts for building a computing workspace ("sandbox"), compiling the code, and running test simulations were written from scratch for both CICE and Icepack; the same scripts are used for both codes to the extent possible. The scripts include specific information for particular computers (platforms or "machines"), and the code and scripts have been ported to numerous machines and compilers at DOE, DOD, NOAA and NSF. Scripts that perform basic plotting of output (time series and maps) were also written to allow "quick-look" analysis of test output.

The testing team developed a strategy for both regular (e.g. nightly) regression testing and testing of new code modifications. Test configurations were developed for full coverage of the column physics code, exercising as much of it as possible; additional tests for other physics available in CICE are being developed now. The Icepack test configurations include various initial conditions (open water, a uniform ice slab, a distribution of different thicknesses, and

land), and forcing (constant values, an annual climatology based on observations, output from the CFS model, and measured values from Arctic and an Antarctic field experiments). We also test for exact restart and have numerous alternative namelist configurations and combinations as part of our test suite. The scripts will set up, run, and report results from the test suites automatically, and a special set of tests are used for continuous integration (CI) of the Icepack test suite through Travis CI, including reporting any failed tests to our test results repository, which is also on github. We tried using CDash for test reporting, and decided that github's wiki offers us more control over the reporting process. Thus our testing is largely automated, although some manual intervention is needed when code modifications are not bit-for-bit. Such changes need to be scrutinized to make sure that they are acceptable. For non-bit-for-bit changes, team members (especially NPS and NRL) designed and implemented a quality control procedure and tests for categorizing the effect of the changes in CICE (discussed below). In particular, the quality control test requirements determined what forcing data is needed, which is now posted on the ftp site.

Since February 2018: Test coverage has been further expanded for both Icepack and CICE with guidance from the codecov.io tool, which analyzes how much of the source code is executed during a set of tests to determine what portions of the code are not exercised. Travis CI was implemented for CICE, with automated reporting to the Test-Results wiki, and code timings were added to track performance changes.

Test reports	https://github.com/CICE-Consortium/Test-Results/wiki

Online documentation

The community support team (NSF) took responsibility for documentation, including technical documents, user guides, and information posted on the wikis. The team developed and implemented a plan for the organization, format, and publication of documentation. Technical CICE documentation for previously released versions of the code existed in a LaTeX document. This was translated into Sphinx, an online package from which html and pdfs can be made, and the parts of it pertinent to Icepack were extracted into the Icepack documentation. The Software Engineer led development of our software development practices documents, including detailed guides for how to fork, commit, pull, push, merge, and work with submodules in git, and prototyped automated generation of our technical code documentation using the readthedocs software package via github. Extensive, additional information is also available on the wikis of all of our repositories.

Since February 2018: All technical documentation was moved to readthedocs.org, and the information on github (wiki and readme files) was reorganized and indexed on a single wiki page (the resource index linked below).

General information wiki	https://github.com/CICE-Consortium/About-Us/wiki
Resource index	https://github.com/CICE-Consortium/About-Us/wiki/Resource-Index
Icepack wiki	https://github.com/CICE-Consortium/Icepack/wiki
Icepack documentation, user guide	https://cice-consortium.github.io/Icepack/
CICE wiki	https://github.com/CICE-Consortium/CICE/wiki
CICE documentation, user guide	https://cice-consortium.github.io/CICE/
Git workflow guidance	https://github.com/CICE-Consortium/About-Us/wiki/Git-Workflow-Guidance

Icepack model development

A new, somewhat simplified driver (based on CICE) was written for Icepack, and scripts and tests were developed specifically for testing the column physics code. The interface for passing variables and data between the column physics code and the driver (or host model, e.g. CICE) was completely redesigned and refactored for improved stability, extensibility, flexibility, and robustness. We had hoped to release Icepack before the Fall 2017 AGU meeting, but decided to spend additional time cleaning up the interface so that host models would not have to be unnecessarily modified in the future (i.e. enabling backwards compatibility in addition to flexibility and robustness). Icepack includes complete test suites for comprehensive code coverage, multiple platforms and compilers (intel, gnu, pgi, nag, clang), and automated testing and test reporting of the main Icepack repository as well as user's forks (on demand) using Icepack's scripts and Travis CI. Documentation is now completely online. The newest Icepack code has been ported to and tested in CICE v6-alpha, and Icepack v1.0.0 was released in February, 2018.

Since February 2018: Expanded test coverage, particularly for the biogeochemistry capability, and other bugs have been fixed. We also obtained a DOI for Icepack: 10.5281/zenodo.1213462

Icepack repository	https://github.com/CICE-Consortium/Icepack
Icepack wiki	https://github.com/CICE-Consortium/Icepack/wiki
Icepack v1.0.0 release with lists of enhancements and bug fixes	https://github.com/CICE-Consortium/Icepack/releases
Sample output	https://github.com/CICE-Consortium/Icepack/wiki/Sample

	-output
Icepack driver design	https://docs.google.com/document/d/1FQISNl-d3YUjZO7XmKVKANguH2FnT9wuB3IEQqdB7ls/edit#heading=h.hyi7l38u12pu
Icepack interface design	https://docs.google.com/document/d/1zK8vvV3KMQ6O7VBaFdDOr4jm6g3gZn7Z8-sDXdxdN8I/edit#heading=h.hyi7l38u12pu

CICE model development

Development of the CICE model through February 2018 by the Consortium primarily centered around needs for Icepack. The code and data directory structure were redesigned for future flexibility, to enable other sea ice model configurations besides CICE to be deposited in the repository. New CICE scripts, most test capabilities, and the CICE test reporting mimic Icepack's. Major new additions to the CICE repository include the quality control testing procedure (developed by NPS), the fast-ice parameterization (developed by the Canadian group), and porting the newest version of Icepack, which involved updating the interface. The quality control tests exploit the statistical properties of sea ice thickness to test both the mean state and the variance, using a 2-stage, paired t-test (which corrects false positives due to autocorrelation) and a quadratic skill compliance test. Grid cells that fail the tests can be plotted on maps using our scripts, for easier analysis.

Since February 2018: Several test suites have been implemented, including numerous combinations of namelist configurations, grid decompositions/parallelization, and smaller suites for fast-turnaround and cloud testing. The fast-ice parameterization has been documented and further improved following suggestions from the user community, and it was used to verify our quality control tests. The quality control procedure was documented in a manuscript submitted to *Philosophical Transactions of the Royal Society A,* now reviewed and returned for revision. Because the code had changed significantly from CICE version 5 and users have been downloading it from github, we created a preliminary "alpha" release of CICE version 6. We continue to enhance and debug this code and scripts. CICE DOI: 10.5281/zenodo.1205674.

CICE repository	https://github.com/CICE-Consortium/CICE
CICE wiki	https://github.com/CICE-Consortium/CICE/wiki
CICE releases with lists of enhancements and bug fixes	https://github.com/CICE-Consortium/CICE/releases

Consortium visibility and community interactions

The Lead Coordinator provided numerous briefings on Consortium activities to various agencies, workshops, and conferences. The Community Liaison has made several presentations advertising Icepack and CICE, and other team members also included visibility for the Consortium in their presentations. A significant amount of the community support team's time was spent on documentation for the codes and our online presence, including a bulletin board and the ftp site hosting our data sets.

2017

- Feb 24 Earth System Prediction Capability Executive Steering Group, Washington DC
- Mar 29 Sea Ice Model Intercomparison Workshop, Bremerhaven Germany (remote)
- May 17 LANL Complex Natural and Engineered Systems climate capability review
- Jun 5-8 ACME all-hands meeting, Potomac MD (poster)
- Jun 19-22 CESM workshop, Boulder CO (poster)
- Jun 26 Interagency Arctic Workshop, Washington DC
- Sep 6 LANL BER Program Review, Department of Energy, Germantown MD
- Sep 11 Isaac Newton Institute for Mathematical Sciences, Cambridge UK
- Dec 13 CICE Consortium Open Meeting during Fall AGU, New Orleans

2018

- Feb 15 NOAA press release
- Feb 17 AAAS Annual Meeting, Austin TX
- Feb 22 Earth System Prediction Capability Executive Steering Group, Washington DC
- Mar 16 contribution to CESM science steering committee presentation, NCAR
- Apr 13 EGU presentation by Rick Allard, Recent Developments in Numerical Earth System Modelling session, Vienna
- May 31 ARCUS Witness the Arctic community highlights article
- DOE ESM program highlight (complete for CICE; Icepack in prep.)
- Quality control publication in *Phil. Trans. Roy. Soc. A* (in review)
- upcoming: IARPC webinar, USGCRP *Our Changing Planet* highlight

Our public meeting in the ARCUS meeting room during AGU was attended by about 25 people, including representatives from many modeling centers that do not currently use CICE. ARCUS requested that we submit an article for the spring issue of their magazine "Witness the Arctic" and suggested that we could also include a "what's happening" update in their weekly email.

To fully test the ridging parameterization in Icepack, we needed forcing data for ice convergence. Cecilia Bitz (Univ. Washington) provided a sea-ice opening/closing dataset from the SHEBA field experiment. New code modifications are beginning to arrive from users outside the Consortium. A particularly intriguing set of changes for improved code efficiency on GPU or KNL architectures has been implemented by colleagues at the Danish Meteorological Institute

(DMI), and our ONR/APPIGO project members are beginning to work with them on this. Elizabeth continues to receive most queries regarding previously released code, as expected.

Since February 2018: We created a zenodo community where all of our releases are collected, gave a presentation at AGU, submitted the QC paper, and provided highlights for ARCUS, NCAR and DOE. Operational uses of the CICE model were also highlighted in a LANL booth at the United States Geospatial Intelligence Foundation (GEOInt) 2018 conference and in a panel discussion during Alaska National Lab Day, Fairbanks AK May 30-31 (both by Cathy Wilson, LANL), as part of LANL's Climate Impacts on National Security thrust. A CICE Hackathon took place in Copenhagen, May 14-18, 2018, with participants from DMI and LANL (Rob Aulwes). Elizabeth Hunke and Tony Craig joined a daily telecon with them to discuss challenges, successes and next steps.

AGU Open Meeting presentation	ftp://ftp.cgd.ucar.edu/archive/Model-Data/CICE/documents/2017OverviewSlides.pdf
NOAA press release	https://research.noaa.gov/article/ArtMID/587/ArticleID/2333/Unique-collaboration-works-to-extend-sea-ice-prediction-from-days-to-decades
DOE ESM highlight	https://climatemodeling.science.energy.gov/technical-highlights/cice-consortium-model-sea-ice-development
Witness the Arctic	https://www.arcus.org/witness-the-arctic/2018/5/highlight/1
Bulletin board	https://bb.cgd.ucar.edu/forums/cice-consortium-model-development
Zenodo community	https://zenodo.org/communities/cice-consortium

Major agency contributions

Agency/ Institution	**Sponsor EOB Member**	**Team members**	**Contributions**
DOE/LANL	Dorothy Koch (Chair) Dave Bader	Elizabeth Hunke Chris Newman Nicole Jeffery Adrian Turner	Project leadership, Icepack configurations/debugging, documentation (especially wikis), BGC, visibility
NSF/NCAR	Anjuli Bamzai Bill Large (EOB Chair)	Dave Bailey Alice Duvivier Marika Holland Bill Lipscomb	Liaison, SE support, documentation design/ implementation, ftp, bulletin board, convergence test,

		Tony Craig	AGU meeting, visibility
DoD/NRL	Dan Eleuterio Ruth Preller	Rick Allard Matt Turner David Hebert	SE support, testing design/ implementation, scripts, plotting tools for analysis, visibility
DoD/NPS	Scott Harper Wieslaw Maslowski	Andrew Roberts	Statistical quality control theory/testing procedure, Phil. Trans. A lead author, aniso rheology updates
NOAA/NWS	Fred Toepfer Hendrik Tolman	Tony Craig Bob Grumbine Brad Johnson Karen Keith	Software engineering (SE), esp. Icepack interface, run/build scripting, test reporting; administrative support, visibility
NOAA/GFDL	Ram Ramaswamy Hendrik Tolman	Mike Winton Anders Dansgaard Niki Zadeh	Travis CI implementation, climatological test for Icepack, code coverage
ECCC	Pierre Pellerin	JF Lemieux Frederic Dupont	Implemented/tested landfast ice parameterization in CICE to calibrate quality control, documented and improved its performance

Plans

Next steps for the Consortium fall into three categories: preparing CICE v6 for release, incorporating community contributed modifications (new parameterizations or capabilities, bug fixes, performance enhancements, etc.), and documenting our quality control testing procedure in a peer-reviewed manuscript (*Phil. Trans. Roy. Soc. A*). Depending on the type and quantity of modifications to Icepack, we also may need to do an updated release to version 1 in the next several months. The Consortium's task list is constantly changing as tasks are completed and new ones arise. Critical tasks for code releases are organized into projects, linked below.

Upcoming changes to the CICE dynamics are primarily contributions from the modeling community that need to be incorporated for CICE v6. Infrastructure tasks include bug fixes and other enhancements designed to make the code more user-friendly. Tasks for the testing and analysis team involve defining and configuring suitable tests to fully exercise the code, and further refining our test reporting process; this work is well underway. CICE documentation

needs to be edited and checked that the Icepack information was properly extracted and referenced; readthedocs now provides a more automated documentation workflow for both CICE and Icepack. In Icepack itself, we will continue work on the biogeochemistry and incorporate available community contributions.

Longer term tasks include enhancing the metrics and analysis capabilities available with CICE, fully automating the test suites and documentation, deciding if and how data assimilation capabilities that have been developed could be included in the Consortium's repositories, and incorporating other community enhancements as they become available. We would also like to organize a workshop for direct feedback and discussions among the broader user community.

New code enhancements expected for Icepack and CICE include a new ridging model, floe size distribution accounting for wave-ice interaction, enhanced snow physics, thermodynamically consistent melt ponds for Icepack, and a C-grid version of the dynamical core, interactive icebergs, and recoding of critical CICE routines to enhance performance on advanced computing architectures such as GPU and KNL. In addition, the Discrete Element Model for Sea Ice (DEMSI), which is being developed at LANL and will use Icepack, is expected to be deposited in the CICE Consortium's repository, but most likely more than one year from now.

CICE v6 project board	https://github.com/CICE-Consortium/CICE/projects/2
CICE issues	https://github.com/CICE-Consortium/CICE/issues
Icepack issues	https://github.com/CICE-Consortium/Icepack/issues
Planning document	https://docs.google.com/document/d/1Nx0EhUw-1BSwnwIn_anCSoaWVLY-kuQHz3MGU5fClp4/edit?usp=sharing

June, 2018 update

Added "Since February 2018" text within sections
Updated or added web links
Minor editing

Trackable Consortium Output	**Identifier**
CICE repository	http://doi.org/10.5281/zenodo.1205674
CICE v6.0.0.alpha release	http://doi.org/10.5281/zenodo.1205675
Icepack repository	http://doi.org/10.5281/zenodo.1213462

Icepack v1.0.0 release	http://doi.org/10.5281/zenodo.1215746
Icepack v1.0.2 release	http://doi.org/10.5281/zenodo.1213463
Roberts et al., *Phil. Trans. Royal Soc. A,* 2018	http://doi.org/10.1098/rsta.2017.0344 (in review, not yet activated)
Allard et al., EGU Abstract	Geophysical Research Abstracts Vol. 20, EGU2018-9495, 2018